

UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS

Real View, LLC.,

Plaintiff

v.

20-20 Technologies, Inc.

Defendant

No. 07-12157

REBUTTAL EXPERT REPORT OF
RANDALL DAVIS

1 July 2009

Contents

I. INTRODUCTION.....	1
II. SUMMARY AND OVERVIEW OF THIS REPORT.....	1
III. ILLUSTRATIVE ANALYSIS.....	6
III A. The Memorandum	6
III B. The Zeldin Affidavit.....	10
III C. Mr. Abbott's Report	20
III D. The Abstraction, Filtration, Comparison Test.....	42
IV. CONCLUSIONS.....	44

I. INTRODUCTION

1. I have been asked by counsel for 20-20 Technologies to comment on the claims and arguments contained in a set of documents I received from counsel, including:

RealView LLC, Boris Zeldin and Leonid Perlov's Motion for Summary Judgment
RealView LLC, Boris Zeldin and Leonid Perlov's Concise Statement of Materials
[sic] Facts in Support of Motion for Summary Judgment
RealView LLC, Boris Zeldin and Leonid Perlov's Memorandum in Support of
Motion for Summary Judgment ["Memorandum"]
Affidavit of Boris Zeldin in Support of Real View LLC's Motion for Summary
Judgment, along with Appendices ["Zeldin"]
List of Alleged Similarities in 20-20 Technologies, Inc's Expert Report
Expert Disclosure of Daniel H. Abbott on Behalf of Real View LLC, Boris
Zeldin and Leonid Perlov ["Abbott"]
[Collectively, "RealView documents"]

2. I have also reviewed a memo from 20-20 Technologies commenting on the Abbott report.

3. This report is based on my experience and expertise in the field of computer science and review of the materials cited in the text.

II. SUMMARY AND OVERVIEW OF THIS REPORT

4. I find that:

- a) The RealView documents attempt to argue, among other things, that the similarities noted in my report of May 5th should be filtered out on the grounds such as "...merger, lack of originality, public domain, scènes à faire, words and short phrases, standard techniques or practices, market/industry demands, considerations of efficiency and compatibility." [Memorandum at 6].
- b) These arguments fail, and an abstraction, filtration, comparison analysis of the two programs demonstrates substantial similarities at the level of detailed expression in the two programs.

- c) These arguments fail in many instances because they do not discuss the similarities cited in my report. Instead they set up carefully generalized straw-man stand-ins, then dismiss these objects of their own creation.
- d) As one example, my report indicates [para 56] that "...there are four ways that both interfaces provide for placement: the Place menu, the relevant icon on the vertical toolbar, navigating through the catalog categories, using the search/Find window" Mr. Abbott's reply, however, addresses itself to the claim that "... both programs *provide multiple means* of placing content in a design" [Abbott at mm, emphasis added], i.e., he has generalized the similarity from the specific set of four that both programs share, to the far broader claim (not found in my report) that there are simply "multiple means." He can then go on to dismiss his version of the claim by saying that "That is true of most CAD programs." [Abbott at mm] Indeed it is true, and this is an effective argument against Mr. Abbott's revised version of the similarity, but it is unresponsive to the claim in my report.
- e) The arguments fail because they at times explicitly ignore expressive aspects of 20-20 software, then claim there is no expression. The selection and arrangement of menu commands, for example, is expressive, yet Mr. Zeldin, in examining the toolbars of 20-20 Design and ProKitchen indicates that "I compare icons on the toolbars based not on their order but on similarities of the underlying features." [Zeldin Exh D, pg 41]. As another example, their analysis of the "Save as Image" command argues that it is a functionality, and uses the curious claim that Google finds that phrase in numerous documents. While the idea of saving the design as an image might be found in other programs, if we look at six other programs for

kitchen design, we find that one is missing the command entirely, while every one of the other five expresses the idea differently (and uniformly) as “Export.” In other words, of a total of eight kitchen design programs, only 20-20 Design and ProKitchen express the idea as “Save as Image.”

- f) Other arguments fail because they must turn to numerous different programs to find – often one by one – all of the similarities shared by 20-20 Design and ProKitchen. As one general example, Mr. Abbott claims that “Nearly all of the individual interface elements in both programs can be found in other CAD programs, both within the narrow kitchen design field and in other more broadly applied software.” [Abbott at 7]. This is somewhat true in one respect – he focuses on the *individual* differences – and illustrative in another sense: even to find the individual items he must expand his search from kitchen design to “more broadly applied software.” This line of argument is akin to claiming that a novel is not original because each of its words can be found somewhere else.
- g) As another example, my report notes that both programs have “...the same sequence of sub-windows: an information box, an edit box, a hierarchical catalog box, a drag and drop listing.” The response in Zeldin [at 15] indicates that “This concept – having sub-windows that show product/folder directories, object location and an image of the product selected before it is placed on the work area is common to many CAD programs, and I have provided numerous examples of this in Exhibit D, pages 4 - 17.” Mr. Zeldin has indeed provided 14 examples of screen layouts, some of which have some of the sub-windows noted in my report, but none of which have the sequence and arrangement of windows found in both

20-20 Design and ProKitchen. His claim and examples thus do not address the expressive elements of 20-20 Design.

- h) Some arguments fail on both grounds: the Zeldin reply cited just above goes on to say that “many CAD programs take exactly this approach,” [Zeldin at 15], which would be relevant, except that by “this approach,” he means simply presenting information about the design in sub-windows (as his examples illustrate). This is of course not the claim in my report; it is instead a generalization of his own invention.
- i) The Real View documents also argue about issues that are irrelevant, claiming, for instance that the sub-windows noted above “...can be selectively minimized, moved to other places on the screen...” [Zeldin at 15]. It is unclear why this should matter in determining whether the screen layouts presented by both programs are similar.
- j) Other arguments appear to invent new theories for filtration. Mr. Abbott replies to the similarity of window layout by noting (at 7) that “The editing screen layout in most CAD software can be customized easily by the user.” This seems to be arguing, remarkably, that the similarity is to be filtered on the grounds that the user can make it go away. While I am not a lawyer, I do have considerable experience with the Abstraction, Filtration, Comparison test, and this does seem to me to be a novel filtration theory.
- k) Other arguments point out differences between the two programs. There are indeed differences, but I understand the issue at hand to be the similarities they share.

- l) Yet other arguments assert claims that are contradicted by very examples the authors cite. Mr. Abbott comments on the similarity in ordering in the vertical toolbars by saying (at para. d) that “It is common to place the tools used most often at the top of a vertical stack, so the tools used most frequently in kitchen design would logically be placed in a similar order.” The problem here is that this is clearly not the ordering principle used in 20-20 Design, yet almost the identical order is found in ProKitchen.
- m) As illustrated below, the Real View documents fail to explore comparable programs and fail to supply examples that illustrate the range of other choices made even by programs within the domain of kitchen design.
- n) As illustrated below, the RealView documents at times cite previous cases inaccurately, resulting in arguments that appear to have no basis.

5. To provide useful, concrete points of comparison for many of the issues discussed below, I refer to six different kitchen design programs, often demonstrating that of the eight total programs, only 20-20 Design and ProKitchen express an idea in a particular way. Those programs are:

Chief Architect X2 Trial Version

Planit Fusion, version 14.2.10

Configura CET

Pro100, version 4.62i demo

KitchenDraw 5.0

Sweet Home 3D version 2.0

6. Of these, 7 appear to be aimed at the same market as 20-20 Design and ProKitchen, namely staff at kitchen design and renovation organizations. Sweet Home 3D is aimed more at the homeowner, but is included as an illustration of comparable software for that marketplace and because it is mentioned in the RealView documents.

III. ILLUSTRATIVE ANALYSIS

7. For the sake of efficiency and conciseness, I discuss below selected illustrative examples of the defects in the three central RealView documents: the Memorandum, the Zeldin affidavit, and the Abbott report.

III A. *The Memorandum*

8. As the Memorandum notes (at 10-11) “...The 20-20 Expert Report identifies as a similarity that in both programs ‘the main window is subdivided into a plan and elevation view.’” Well, almost. The Expert Report actually says “In both the main window is subdivided into a plan and elevation view, with the plan at the bottom and the elevation at the top.” In other words, the Memorandum has cited only the beginning of the claim, generalizing it, so that they can argue against the broader version.

9. They do so by noting “As Mr. Zeldin shows, this feature, which is sometimes called ‘multiview,’ is common to many CAD programs. (Zeldin Aff., par. 23-32).” [actually Mr. Gesmer means Zeldin Exh D, pp 23-32]. Indeed, *some* collection of multiple views is a common feature, but the claim was not the generic notion of dividing the screen vertically, nor of simply “having multiple views.” It concerned the one choice that both programs share, namely having the elevation at the top and the plan at the bottom.

10. Zeldin Exh D provides five screen shots of other programs that offer multiview. Interestingly, two of those don’t split the screen between elevations and plans at all (they show

plan and 3-D views), and none show a split screen with elevation above plan. Those that do contain elevation and plan views show the plan on top. Even Mr. Abbott inadvertently agrees – his (literally) textbook example shows “...the way a draftsman would have laid out a sheet when drawing by hand” [Abbott at 11], and has the plan at top and elevation at the bottom.

11. In other words, among all of the examples of cited in all the RealView documents, *only* the screenshots of 20-20 Design and ProKitchen show a screen split with elevation at the top and plan at the bottom, the similarity noted in my report. Hence RealView’s own examples make the point convincingly that the two programs share an uncommon expression of interface design.

12. The Memorandum goes on to say that “Moreover, there is only one way to present this feature (a horizontal screen split). (Zeldin Aff., Ex. D, p. 25).” This is contradicted by Planit Fusion, a design program not discussed by the RealView documents, that provides a feature called precisely “Multi View,” which splits the screen both horizontally and vertically, into four quadrants. Apparently there is indeed more than one way to present this information.

13. The Memorandum also attempts to address the issue of “construction line” and its uncommon use as a way of expressing a kind of wall. RealView’s own expert agrees that “the use of the term ‘construction line’ to represent a phantom wall is unusual in CAD software,” [Abbott at 26]. The best defense he can find is that “using the term ‘construction line’ to mean a phantom wall would not be confusing to a professional user, nor would it present any difficulty to a new user.” [Abbott at 26]. So RealView’s own expert agrees that the use of construction line to represent a phantom wall is “unusual,” an unusual usage found in both 20-20 Design and ProKitchen.

14. The Memorandum looks to a previous case to argue that a concept similar to a construction line had previously been found unprotectable, saying that “Mitek Holding, Inc. v. Arce Engineering Co., 864 F. Supp. 1568 (S.D. Fla 1994), *aff’d*, 89 F.3d 1548 (11th Cir. 1996), is instructive on the ‘construction line’ issue.” [Memorandum at 13] The Memorandum continues:

There, the court was faced with a software program and copyright claims similar to this case. The suit involved the user interface of “layout program” that drew “architectural blueprints” – in other words, a CAD program. In evaluating a list of 18 alleged similarities submitted by the plaintiff, the District Court held that the “expression of work places without reference to the real walls, the roof , or the ceilings” (which sounds like a “construction line,” although the court did not use that term) was an unprotectible idea. 864 F. Supp. at 1583.

15. The Memorandum’s citation from the case is instructive, though not perhaps in the way intended. As noted above, it quotes the case as saying “...the District Court held that the ‘expression of *work places* without reference to the real walls, the roof , or the ceilings’ ... was an unprotectible idea” [emphasis added]. The Memorandum also suggests that the phrase *work places* “sounds like a ‘construction line.’ ” Perhaps it does, but this is not what the Court actually said.

16. The Court was analyzing a claim about “Expression of work *planes* without reference to the real walls, the roof, or the ceilings of the building.” [emphasis added]. It’s clear that the word is *planes* rather than *places*: the software in question is used to design roofing structures, and it handles the sometimes difficult issue of the geometry of multiple intersecting roof planes. This is evident in the remainder of the court’s discussion of this element, and is something I can speak to from personal experience, as I was an expert for Mitek in this case and spent substantial time studying both the task and the software in question.

17. Also as noted, the Memorandum also cites the case for the proposition that

...“expression of work places without reference to the real walls, the roof , or the ceilings” (which sounds like a “construction line,” although the court did not use that term) was an unprotectible idea. 864 F. Supp. at 1583. [Memorandum at 13]

18. Yet what the Court actually said about this item (#10 on its list) was

10. Expression of work planes without reference to the real walls, the roof, or the ceilings of the building. This feature makes it easier to draw layouts of complicated structures.

As discussed supra, the Court finds that the use of intersecting planes is a process that is not entitled to copyright protection. However, although the doctrine of merger denies copyrightability to the depiction of planes as trapezoid shapes, the same cannot be said about the depiction of planes without reference to the connecting walls, ceilings or roof. The Court finds this is an original aspect of the Aces Layout Programs and is therefore entitled to copyright protection.

19. It is not evident how the Memorandum extracts from this support for the notion that work planes were found to be an unprotectable idea.

20. The Memorandum at times also generates its own reinterpretation of claims in my report. As one example, it suggests [at 18] that

The 20-20 Expert Report claims similarities based on the use of “sub-windows” or “panels” on the left side of the screens of the two programs when the program is first installed (“out of the box”). (Item 14, Expert Report par. 34, 35). As shown in the Zeldin Affidavit, many CAD programs use panels from which the user can select objects necessary to the creation of a design. (Zeldin Aff., Ex. D, pp. 4-17). Thus, the “look” of the two programs is industry standard for computer aided design software.

21. My report does not claim simply that there are panels on the left side of the screen, saying instead [at 21], “In both cases the left side of the screen contains the same sequence of sub-windows: an information box, an edit box, a hierarchical catalog box, a drag and drop listing.” In other words, the issue is a particular set of windows, in a particular selection and arrangement. The Memorandum attempts to ignore this, allowing it to focus instead on non-issues (e.g., “CAD programs use panels”) and make broad unsupported claims, such as “The

‘look’ of the two programs is industry standard.” This is true only if by “look” we mean something like “any use of sub-windows for any purpose in a design program.” And this is manifestly not what my report claimed as the similarity.

III B. *The Zeldin Affidavit*

22. Mr. Zeldin frequently argues in broad generalities about what users expect to see and common conventions in the field. Unfortunately he often does so when a review of six other kitchen design programs would demonstrate that his claims are simply untrue.

23. As one example, Zeldin indicates [at para 37]:

In many instances the sequence of commands is determined by the logic of a CAD software application. Windows users, and more specifically CAD users, expect to see a certain “logic” to a CAD program, just as the user of a word processor expects to see familiar commands in familiar places. For example if one wanted to put a cabinet or chair in a design, convention suggests that the user go to the “Place” command on the main menu, where the user would be presented with a choice of objects to “Place” in the design. Much of a CAD program follows similar conventions, and therefore while there are some choices as to how to organize commands, the choices are often limited by convention and logic – commands must go where users expect to find them. (see Ex. D, p. 34 for the “Place” menu).

24. Consider in particular the claim that “...if one wanted to put a cabinet or chair in a design, convention suggests that the user go to the ‘Place’ command on the main menu, where the user would be presented with a choice of objects to ‘Place’ in the design.” We consider six other kitchen design programs to see how each one places a cabinet or chair.

- a) Planit Fusion Live: there is a Place menu, but it cannot be used to place cabinets and chairs. To place a cabinet you click on the + to display the Add Palette, go to the Add Palette and click to select the right category of item (e.g., cabinet), click on the correct item, then place it on the design.

- b) Chief Architect: there is no Place menu. There is a Build menu, which has the ability to add cabinets and other items to the design and is similar to the Place command in 20-20 Design and ProKitchen.
- c) Sweet Home: there is no Place menu. There is instead a hierarchy of items that can be traversed to find the relevant object (e.g., cabinet), which can then be added to the design.
- d) KitchenDraw: there is a Place menu, but it cannot be used to place cabinets, etc., offering only a Wall placement command. Other objects are added to the design by traversing the object hierarchy at the right of the screen.
- e) Pro100: there is no Place menu. Objects are added to the design using the Edit menu, then the Insert from Catalog menu item.
- f) CET Designer: there is no Place menu; items are added to the design by selecting them from the palette at the left of the screen.

25. In other words, despite Mr. Zeldin's claim that "...convention suggests that the user go to the 'Place' command on the main menu, where the user would be presented with a choice of objects to 'Place' in the design," four of six competing kitchen design programs do not even have a Place command on the main menu, and the two do have it do not offer the ability with that command to place anything but walls. Only Chief Architect offers a command (Build) that is similar to Place in 20-20 Design and ProKitchen.

26. Simply put, while all of the competing programs offer the function of placing items in the design, only *one* comes close to expressing it the way 20-20 Design and ProKitchen do. Of those eight total programs, only 20-20 Design and ProKitchen express it the way described in my report as a similarity, with some similarity to Chief Architect.

27. Consider next Mr. Zeldin's claim that [para 47] that:

The information contained in each of the panels in ProKitchen is essential to the application: users must be able to use the file structure to locate products; users must be able to see a preview of the product before they select and place it; users must be able to see the precise location of a product as they place it within a design; and, users must have the ability to select a product from the catalogs using a search field.

28. Note first that this claim is about "the information contained in each of the panels," not the manner in which that information is expressed. Hence Mr. Zeldin's claim is, as I understand it, fundamentally irrelevant to the question of similarity of expression.

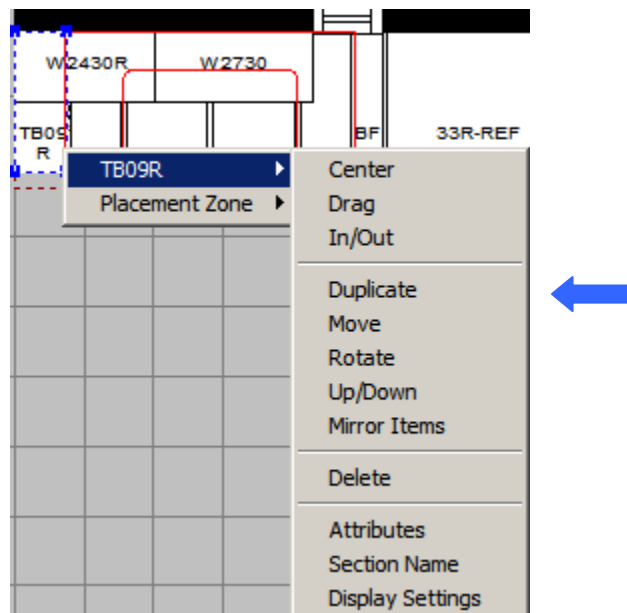
29. Even so, considering it for the sake of argument, a brief review of the six competing kitchen design programs makes several things clear. First, while there are some individual similarities – KitchenDraw's Articles panel presents an item hierarchy and preview picture, as does Chief Architect's Library browser window – even here there are differences in expression (panels on the other side of the screen, preview images at the bottom, not the top), making it clear that there were alternatives expressions available to ProKitchen even at the level of these details. Second, none of the other four programs have a selection and arrangement of panels that is at all similar to 20-20 and ProKitchen. This too indicates that, had RealView wanted to, it could have expressed the desired information in any of the very different ways found in the other programs.

30. Mr. Zeldin claims [at para 27] that tabbed interfaces are routine, but once again this is a generalization of the actual similarity found in 20-20 Design and ProKitchen. In addition to the similarities described in my initial report, there is the issue of the "layering" available in the tabs of both 20-20 Design and ProKitchen, but in none of the other competing programs. In both 20-20 Design and ProKitchen, the user can decide which elements ought to appear on each tab. Using almost identical expression, both programs allow the user to decide, for instance, to show

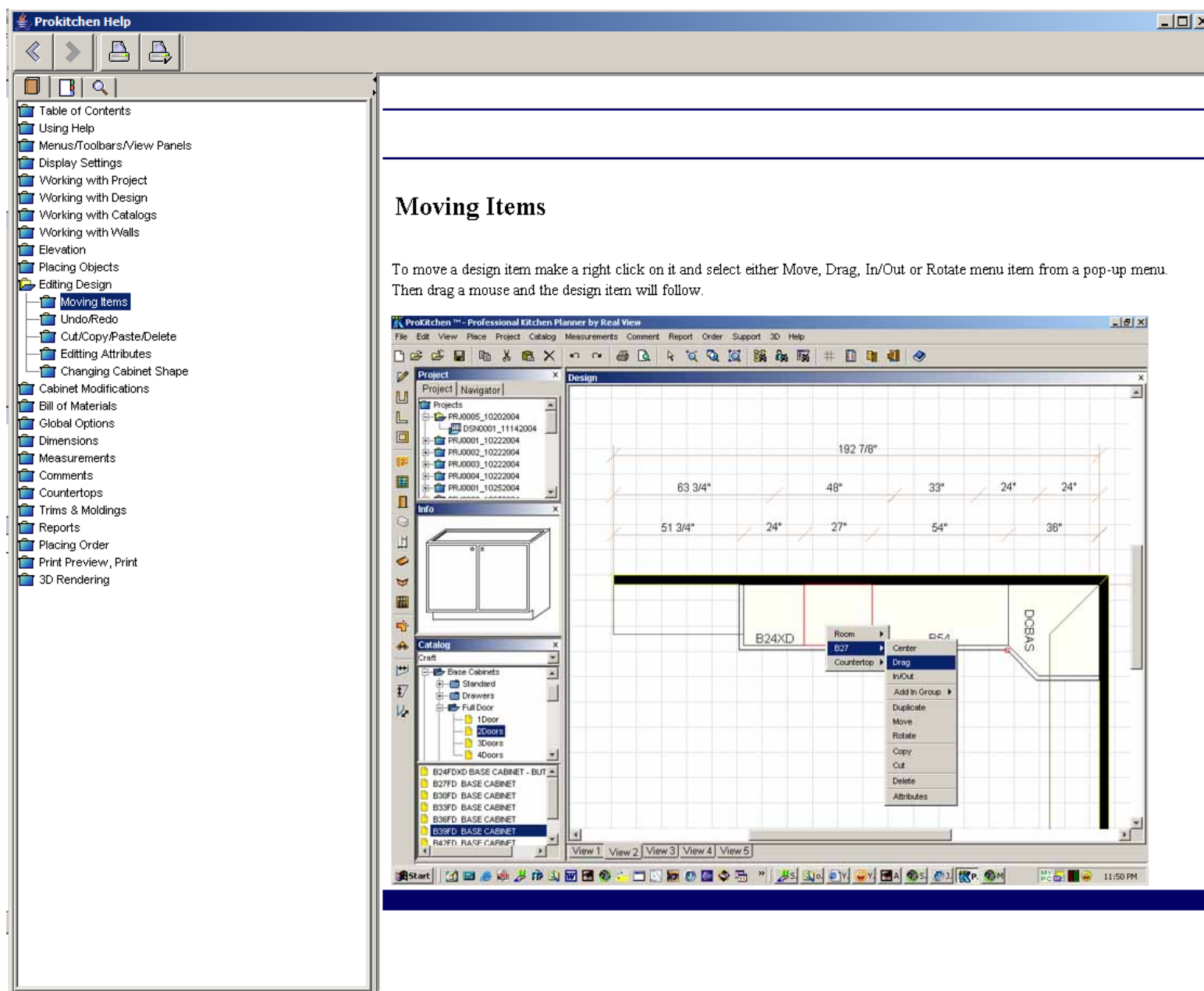
only the walls and cabinets in one tab, only the walls and appliances in other tab, etc. Yet once again, none of the other competing six programs offers anything like this selective display capability oriented around kitchen design elements.

31. Mr. Zeldin also claims [at para 33] that ProKitchen “has not copied any of the menus in 20-20 Design,” and suggests that “Place is the only menu item in common” in the main menus. More accurately put, Place is the only menu item that is verbatim identical in the two menus; as my report showed, there is substantial similarity in the selection, arrangement, and expression of information in the Notes/Comments and Dimensions/Measurements menus of the two programs.

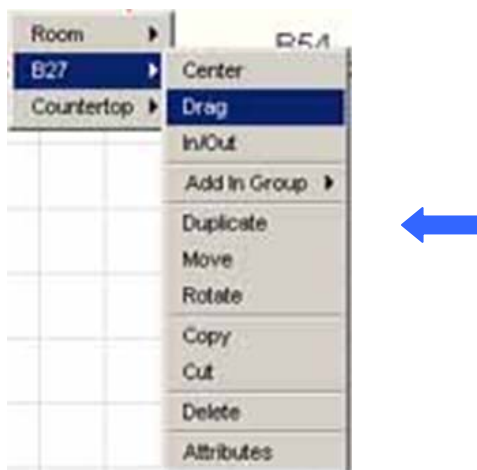
32. Mr. Zeldin’s disclaimer of copying 20-20 Design menus is also undermined by the Help files found in ProKitchen 2.2.0. The 20-20 Design program has a command to Duplicate an object in response to a right click on it:



33. ProKitchen has a similar *functionality*, i.e., the ability to duplicate an item, via its Edit, Copy menu items, yet, curiously, the Help file for ProKitchen 2.2.0 contains the following illustration:

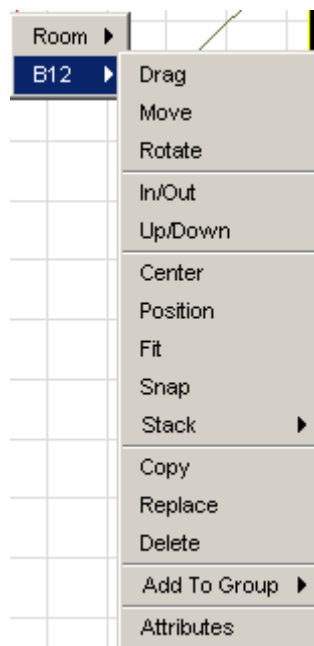


34. Zooming in on the context menu that appears in that figure,



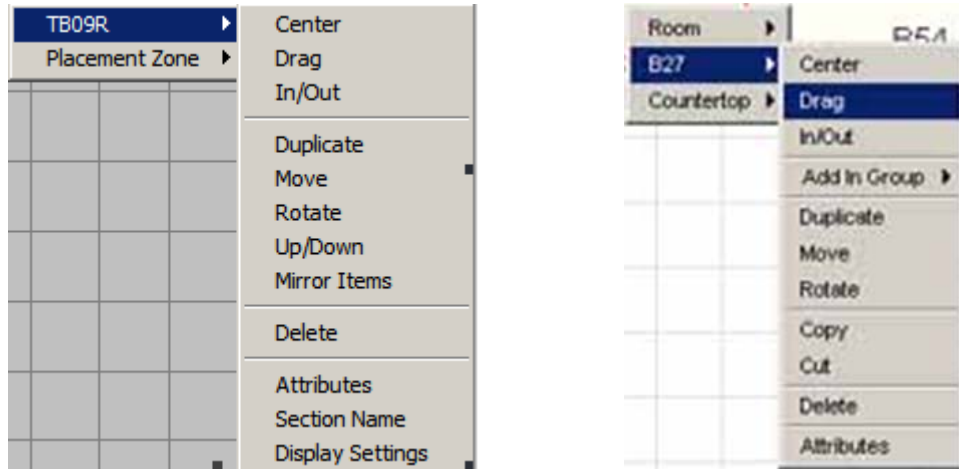
we can see that the Help file in ProKitchen 2.2.0 indicates that the program contains a Duplicate command in the context menu. This is not true, there is no Duplicate command in the program.

Here's the context menu that appears in actual use:



35. Simply put, the Help files in ProKitchen 2.2.0 advertise a context menu that (a) is different from the one in that actually appears in use, (b) that contains a Duplicate command that

appears nowhere in the program, and (c) is in fact similar to the actual context menu from 20-20 Design, version 6.1. Here, side by side, are the context menu for 20-20 Design and the illustration from ProKitchen 2.2.0 Help:

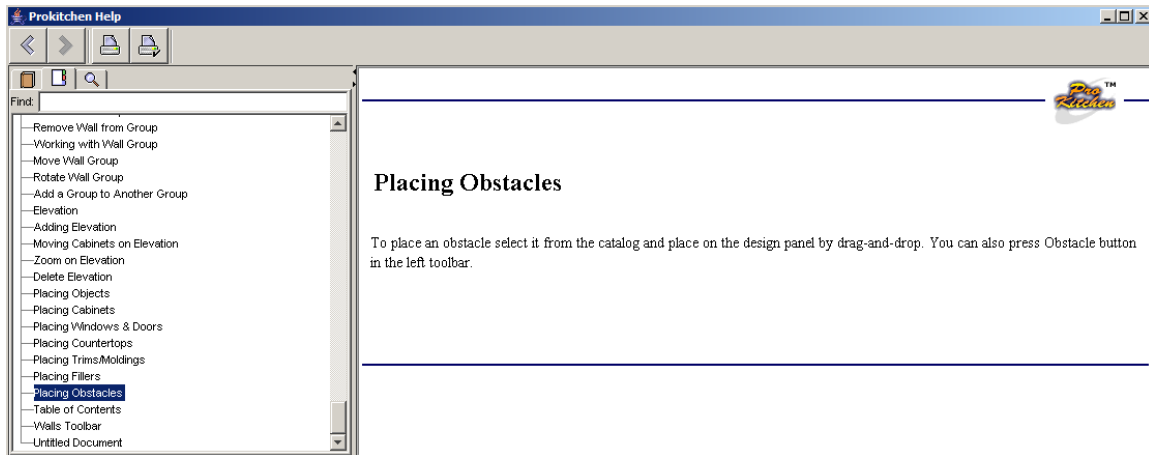


Context menus for 20-20 Design version 6.1 (left) and Help file from ProKitchen 2.2.0

36. Note the substantial overlap in selection and arrangement: both have identical names in the same order: Center, Drag, In/Out, Duplicate, Move, Rotate, Delete, Attributes.

37. The most plausible explanation for this is that RealView wrote the Help file as a sort of design document, and they did so by copying 20-20 Design's menus quite closely. They later evidently changed the actual design of the program, but left behind the evidence of their copying in the Help files.

38. As another example, 20-20 Design has the ability to place what it calls Obstacles in the room and an icon/button for this in the vertical toolbar. As Mr. Abbott indicates "The use of the phrase 'Openings & Obstacles' as used in 20-20 Design is unusual – possibly unique – and doesn't appear on the ProKitchen menu." Interestingly, while it does not appear in ProKitchen menus, it does appear in ProKitchen 2.2.0 Help:



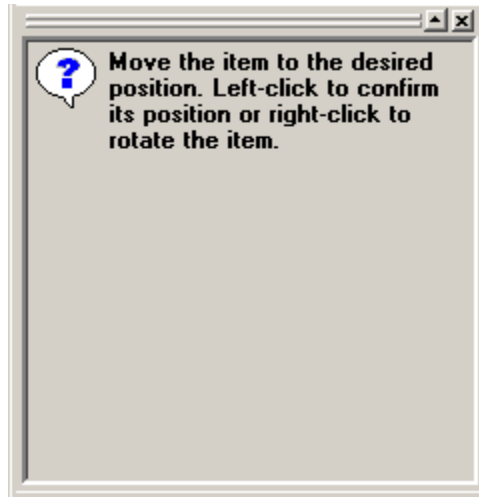
39. The Help file says that “You can also press the Obstacle button on the left toolbar.”

Alas, there is no Obstacle button on the left toolbar, at least not in ProKitchen. There is such a button in 20-20 Design.

40. The Help file in ProKitchen 2.2.0 labeled Info Panel claims that [emphasis added]:

The Info Panel displays *both the images of the catalog items and info messages with the instructions on the performed tasks*. The Info Panel is accessible from View/Info menu item. It can be resized by dragging the panel's border up or down. You can close it by pressing on x button at the right upper corner.

41. This, too, is not true for ProKitchen, but is true for 20-20 Design. 20-20 Design first shows an image of the selected item, then replaces the picture of the item with what might accurately (if somewhat ungrammatically) be described as “info messages with the instructions on the performed tasks”:



42. Interestingly, ProKitchen version 3.0 changes the Help text for this item, saying:

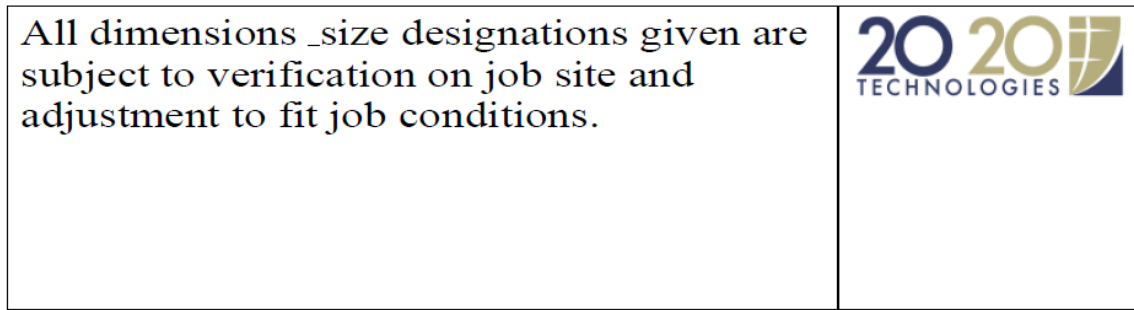
This panel allows you to view what the selected cabinet looks like before placing it on the floorplan. You have the option of viewing the cabinet in a black and white image or in a fully-rotational color 3D view. Please note: You can change between the two viewing options [sic] by double-clicking on the image in this panel.

43. In other words, ProKitchen 3.0 Help describes what ProKitchen software actually does, while the 2.2.0 help describes 20-20 Design.

44. So while Mr. Zeldin claims that "...RealView has not copied any of the menus in 20-20 Design," the evidence is clear that they did, and they did it so thoroughly that some of the Help files in ProKitchen 2.2.0 are a more accurate description of 20-20 Design than they are of ProKitchen.

45. In addressing the question of the Legend stamp, Mr. Zeldin claims (para 69) to have "...provided numerous examples of similar or identical 'legends' that are available on the Internet. (Ex. D, p. 71-75)." There is one problem with several of his examples of identical legends (Zeldin Exh, pages 72 and 73) – they are instances of 20-20 software, specifically they show legends produced by 20-20 Design version 6.4, evidently when it was licensed to other users.

46. The evidence for this is clear if we examine a close-up of the legend created by 20-20 Design, version 6.4, and note the spurious underbar character preceding the word “size”:



47. The legend produced by version 6.4 had this mistake in it. A close examination of four examples cited by Mr. Zeldin beginning on page 72 of his Exhibit makes it clear that they, too, have this spurious character, and could only have come from the 20-20 Design software.

48. Hence the point remains that only 20-20 Design and ProKitchen version 2.2.0 share the same mistake in the wording of the legend, making it clear how diligently RealView programmers copied 20-20 software.

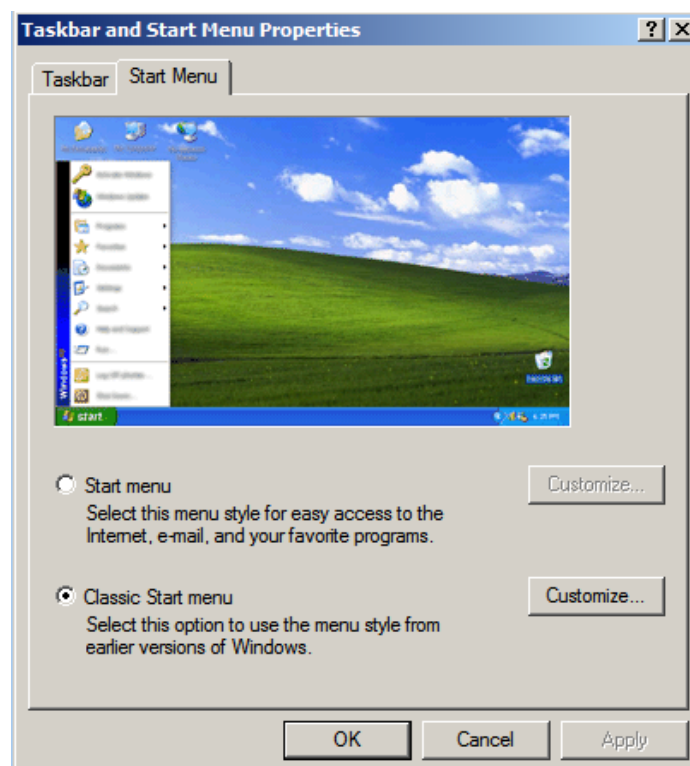
49. Mr. Zeldin also claims (para 69) that the text can be changed. Indeed it can, but the question here is not whether the user can make the programs express themselves differently, it's whether RealView produced a program similar to 20-20 Design, which it did.

III C. *Mr. Abbott's Report*

50. Mr. Abbott suggests [at 4] that:

The interface of a CAD program is not a significant part of a company's decision to purchase one program over another. Those decisions, in my experience, are based on four criteria: features, cost, support, and the availability of trained users. The interface plays almost no part in adopting a piece of software. People who “place increasing emphasis on how easy and intuitive” programs are to use are likely to be homeowners, not professionals. Professionals will take the time to learn software if it is useful to them, regardless of the interface.

51. I disagree and moreover find the argument circular. First of all, the interface is a very important part of usability of software; at MIT, for example, there are two different courses in the computer science curriculum devoted solely to user interface design and implementation, both of which stress design principles that make software easy to use. Second, the importance of familiarity in usability is so widely recognized that even Windows XP offers all users a choice of the interface for the Start menu, explicitly allowing them to “use the menu style from earlier versions of Windows,” even though it claims that its newer version provides “easy access:”



52. Next, even Mr. Zeldin disagrees with Mr. Abbott. The Zeldin affidavit is very clear on the issue: “commands must go where users expect to find them” [Zeldin at 37]. Mr. Zeldin

understands the significance of the user interface; if commands are not where the user expects, the software is more difficult to use.

53. Mr. Abbott's argument is also self-contradictory. He denies the significance of the user interface, while acknowledging the importance of "the availability of trained users." One of the most valuable kinds of training one can have is in the interface for the specific software in question. If I know not just general principles of CAD software, but the particular expressive design choices that have been made for the interface of a specific program like 20-20 Design, I can be far more effective and efficient. Hence Mr. Abbott actually acknowledges the value of interfaces, but hides it behind the notion of trained users.

54. I understand that 20-20 Technologies has historically expended substantial effort in educating its user base about the interface, including requiring new purchasers of 20-20 Design to take a half-day training course given by authorized consultants. The company has thus invested in and produced a substantial population of "trained users" because of the value of that interface.

55. Mr. Abbott also does not seem to understand that the market for 20-20 Design is largely computer-aided sales, not computer aided design. That is, the intended user audience is people like the staff at stores ranging from Home Depot to small kitchen retailer showrooms, people who are kitchen designers and builders, not CAD professionals. For this market, the question of what a professional CAD user might do with software is basically irrelevant.

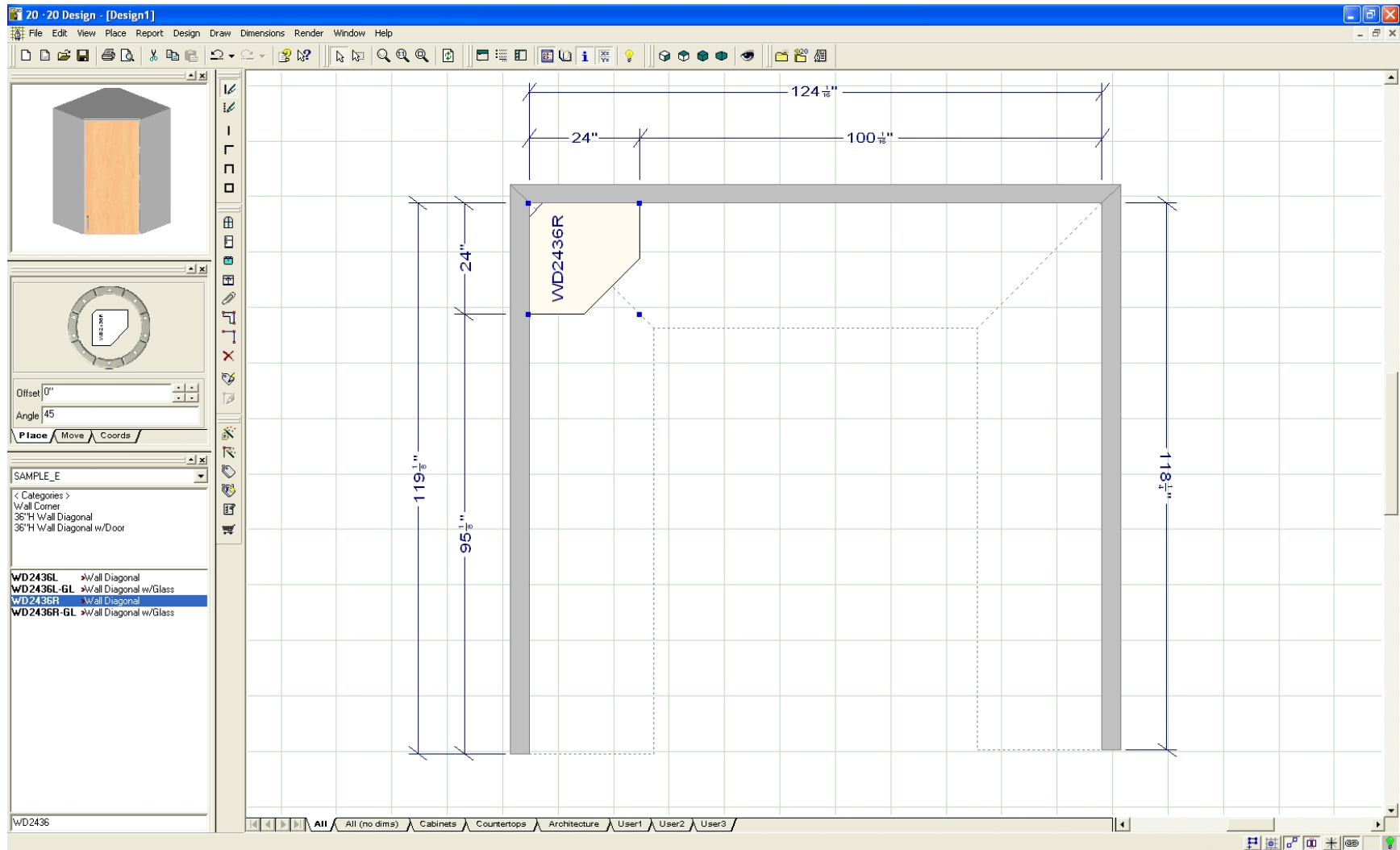
56. Mr. Abbott suggests [pp 6-8] that users see splash screens before encountering the primary user interface. This is true, but such screens appear fleetingly, and are in any case rarely shown during demonstrations at trade shows, where the focus is on actual use of the program. Hence prospective customers are unlikely to be exposed to the splash screens.

57. Mr. Abbott argues [at 6-10] that the layouts of ProKitchen and 20-20 Design can be changed by the customer, but this is irrelevant. The question is whether ProKitchen has copied protectable elements of 20-20 Design, not whether the customer is given the ability to hide that copying.

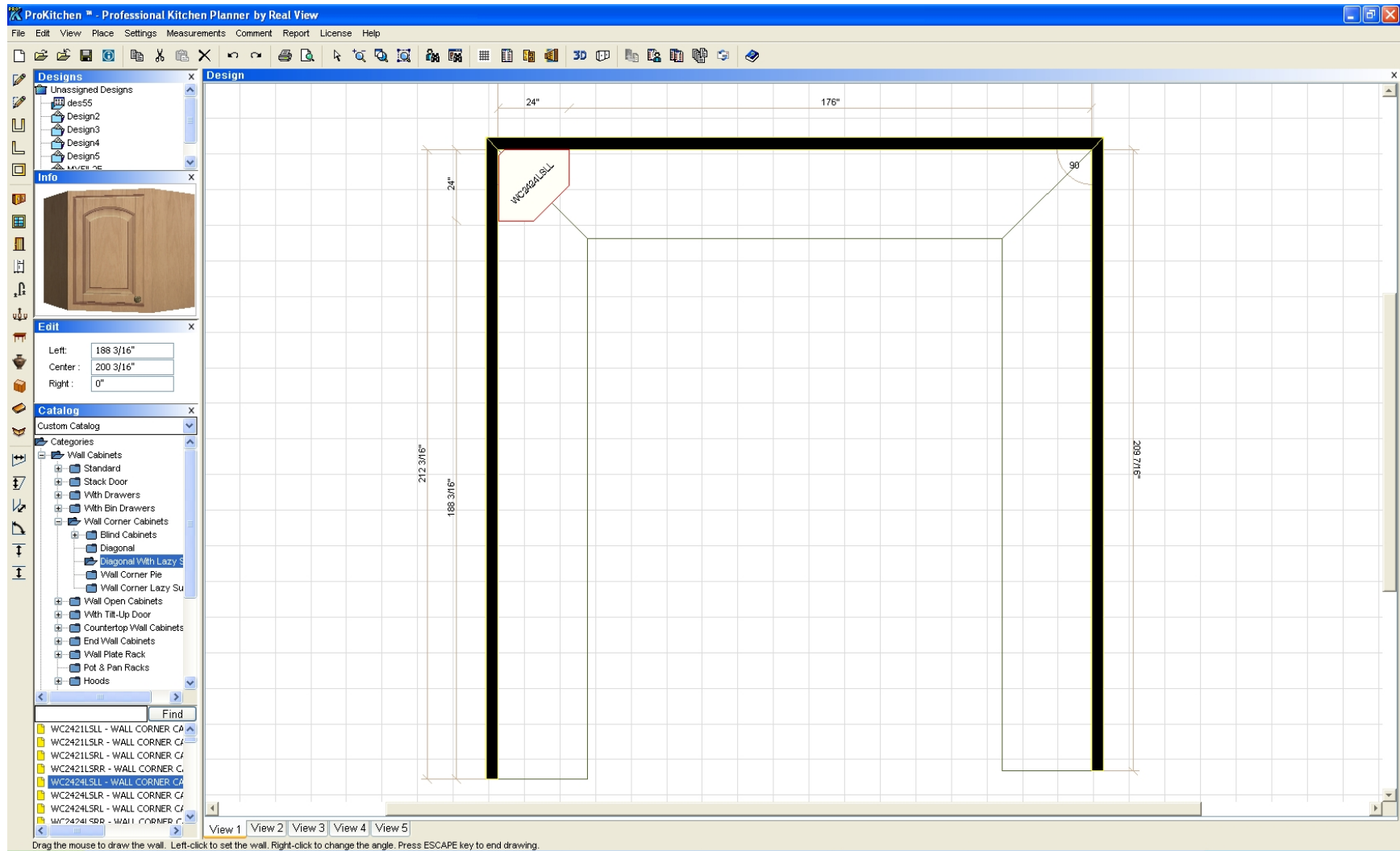
58. The Abbott report also claims [pp 8-10] that the two programs in question “share elements with many other applications.” Again, even if this is true, it is not the issue at hand. The question is not whether each one of the elements in the interface of 20-20 Design can individually be found elsewhere; the question is whether the selection and arrangement embodied in that program has been copied by ProKitchen.

59. Mr. Abbott addresses this in passing by offering screen shots of four programs (KitchenDraw, Chief Architect, SolidWorks, and AutoCad), without further comment. For a more detailed comparison, I present below screen shots from all eight of the kitchen design program, showing each as it looks when a new design has begun, with three walls defined and the first cabinet (or other object) placed. (Having something in the design offers a more detailed comparison.) The similarity between the overall layout of 20-20 Design and ProKitchen is clear; the relative dissimilarity with the other programs is also clear.

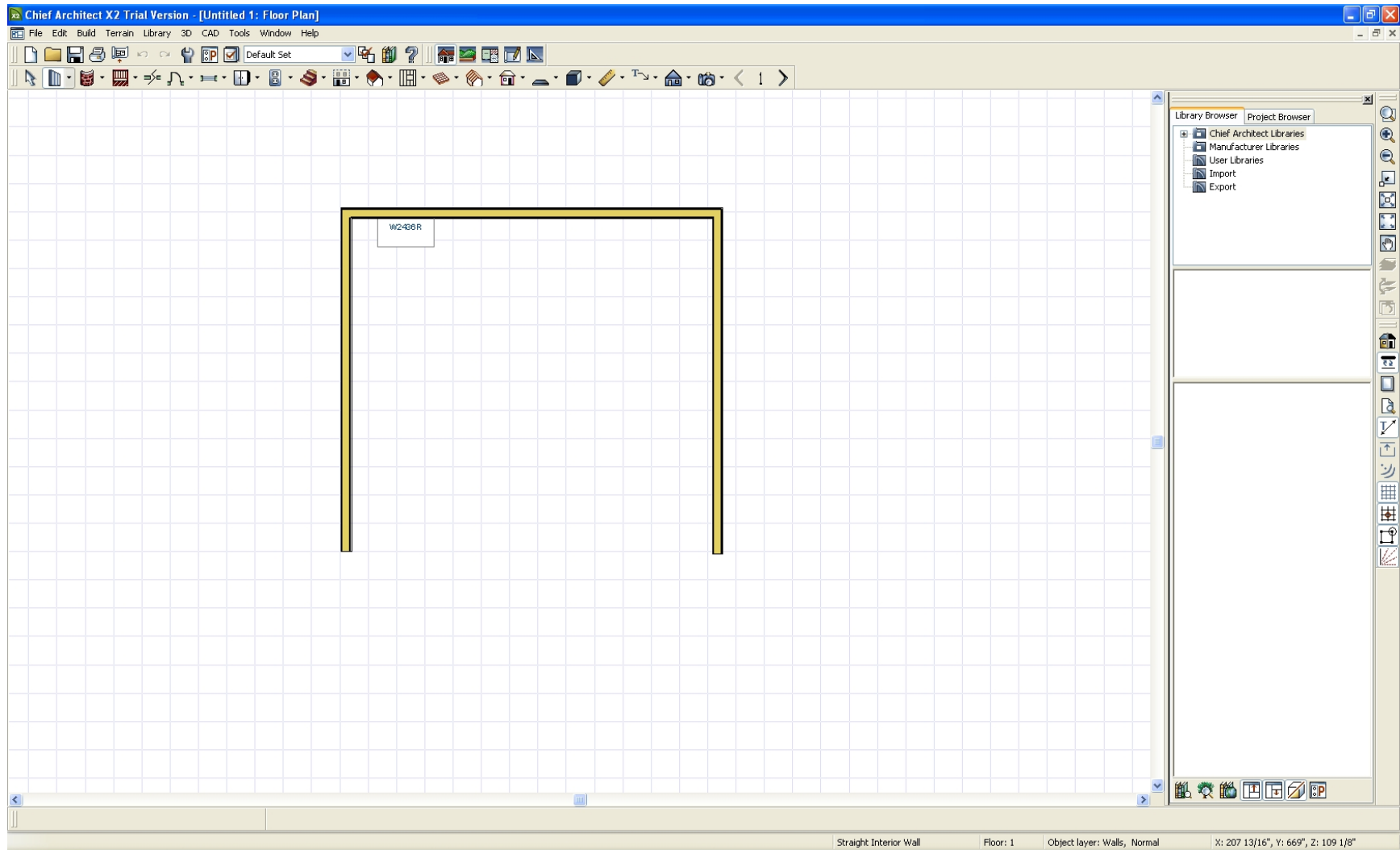
20-20 Design, Version 6.4



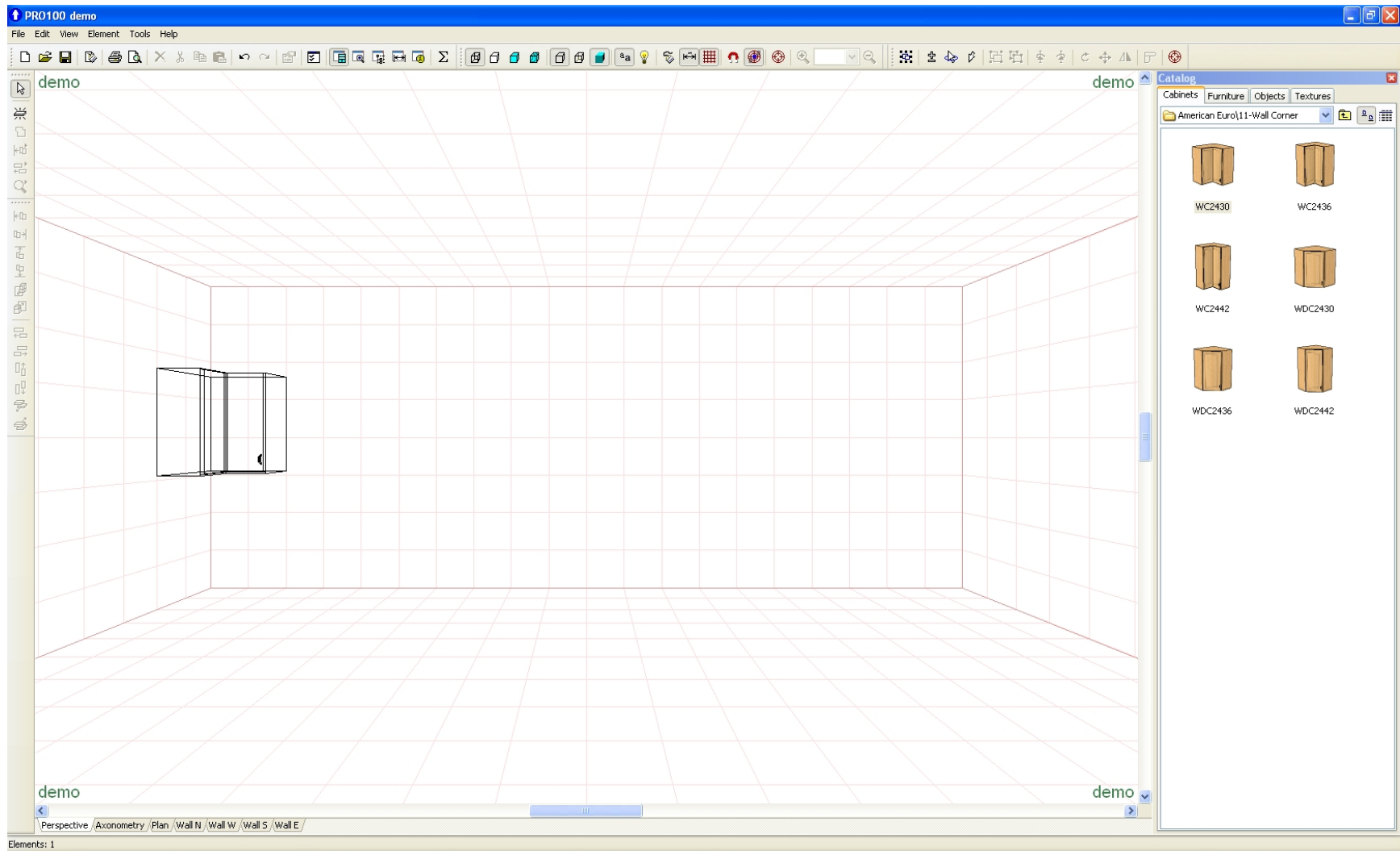
ProKitchen, Version 3.0



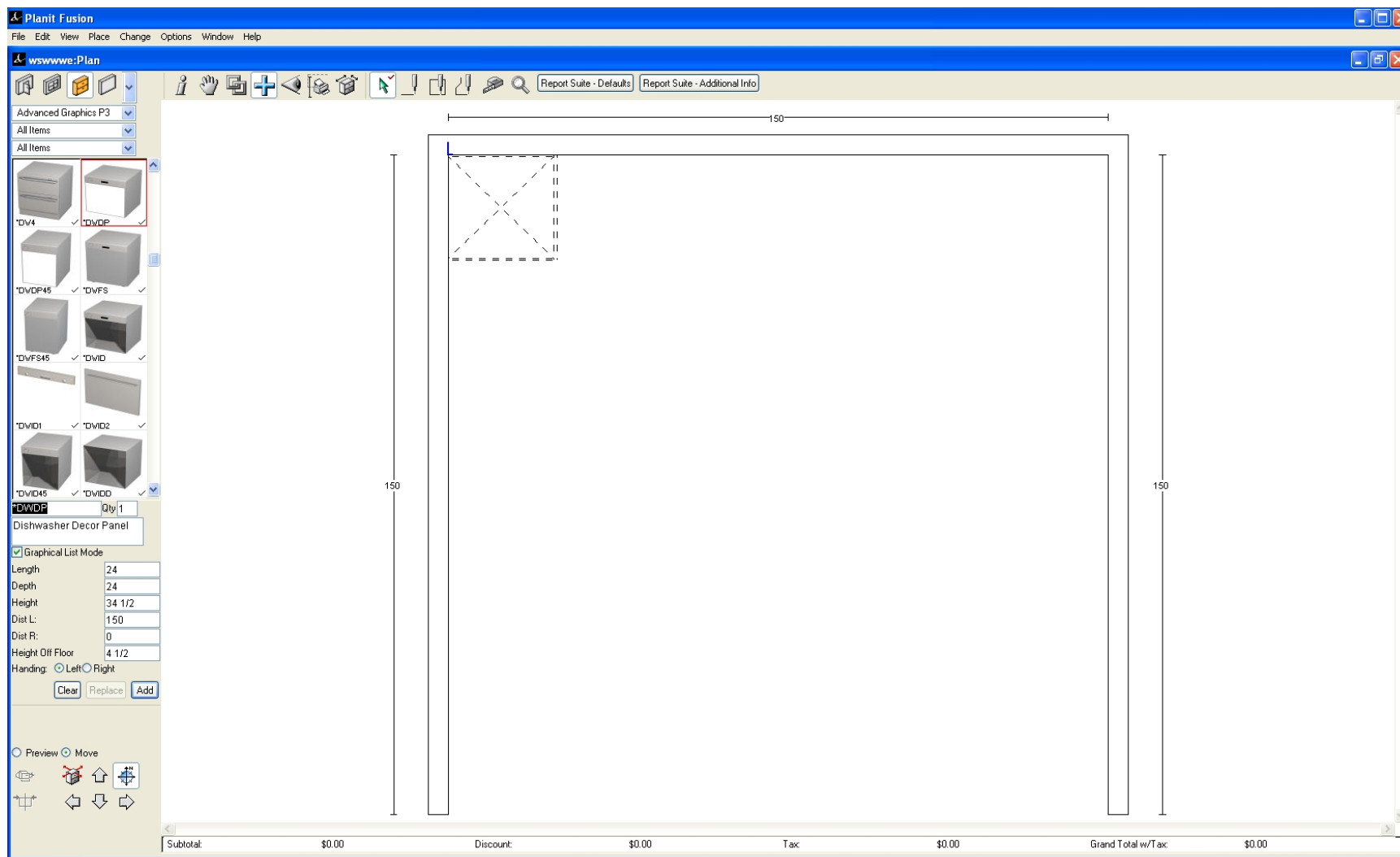
Chief Architect



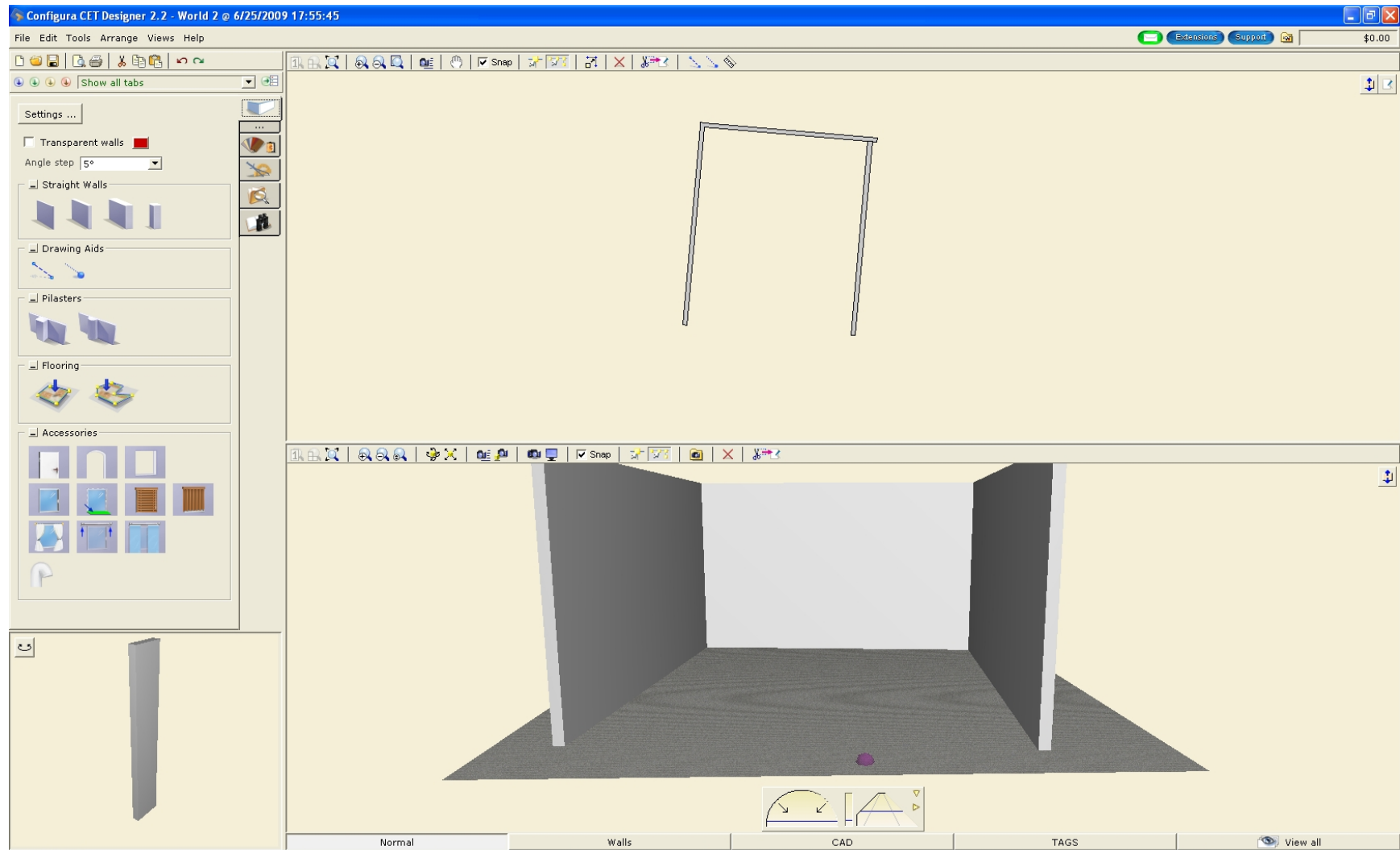
Pro100



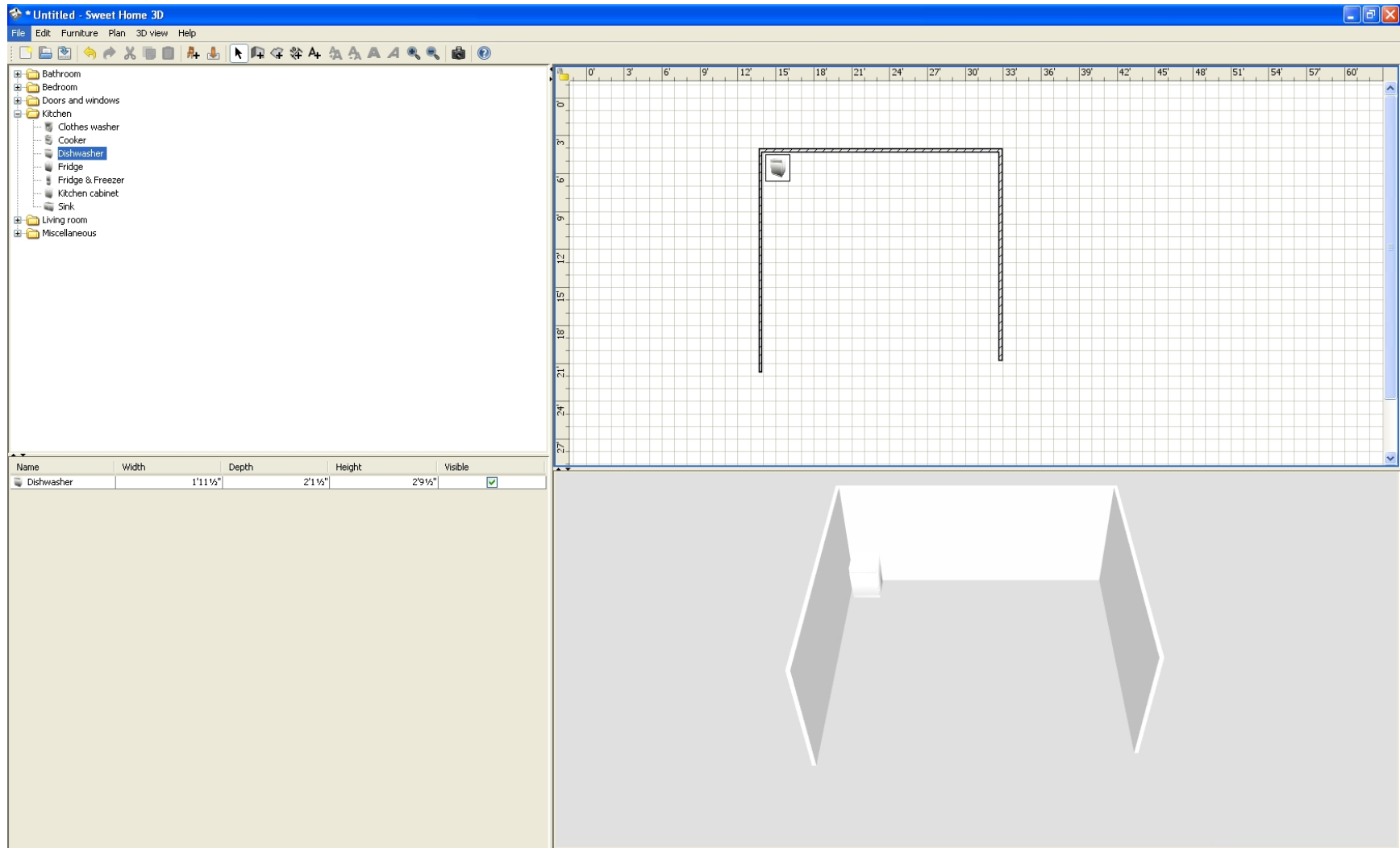
Planit Fusion Live



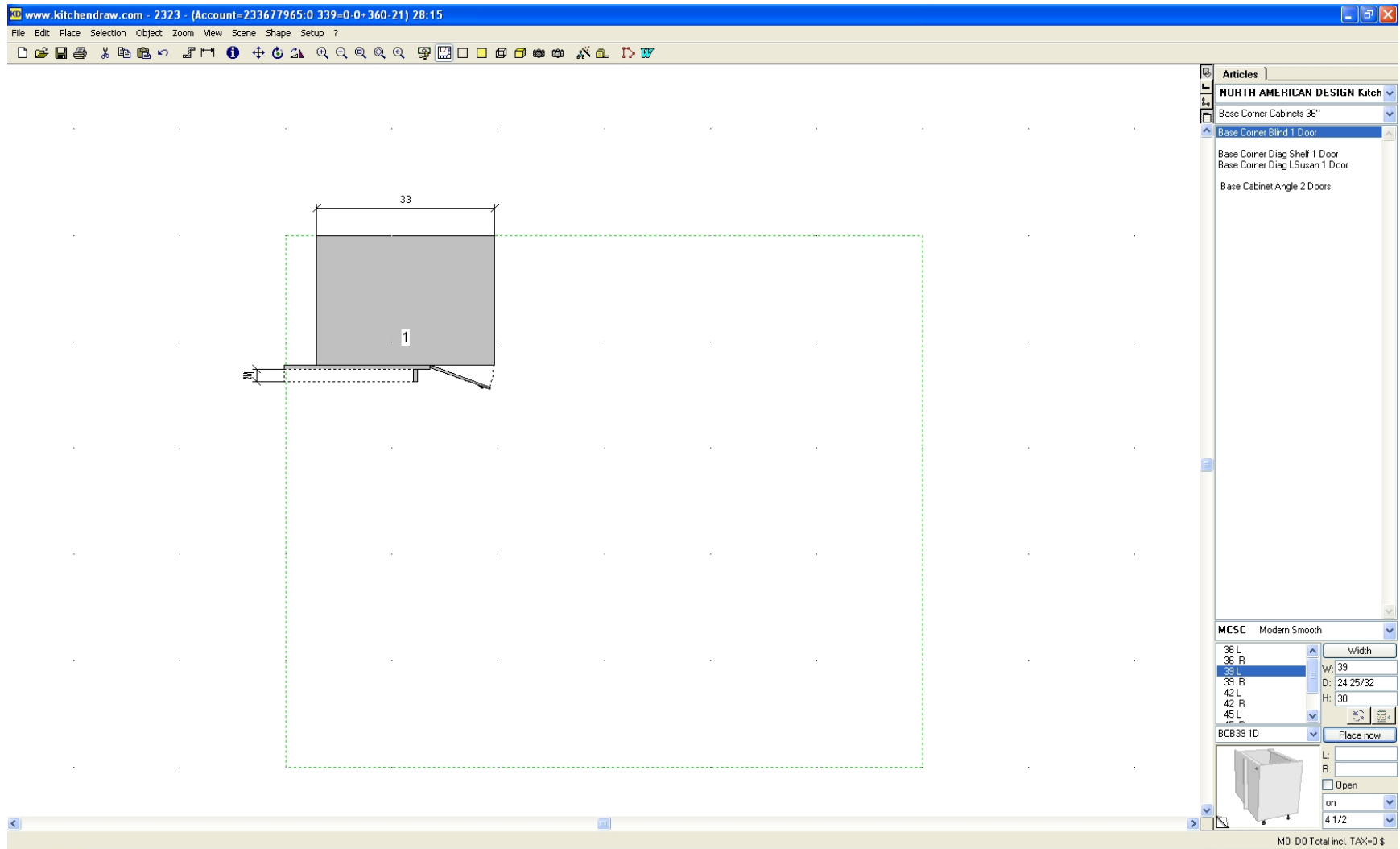
CET Configura



Sweet Home



KitchenDraw 5.0



60. At page 11, Mr. Abbott notes that “Placing panels or windows on the left side of the screen is common, but arbitrary. They could have been placed on the right without changing the function of the software.” Yes, just exactly so. The choice is one of many: the panels could have been designed differently, placed differently, the function of the panels could have been expressed in mechanisms other than panels, etc. As Mr. Abbott agrees, even this small choice was arbitrary, i.e., not required by the task, and thus is one small element of expression in 20-20 Design.

61. Mr. Abbott also notes (at 11) that

Vertical toolbars containing icons are standard in design software. All of the software illustrated above in section V.a. of this document have vertical toolbars. It is common to place the tools used most often at the top of a vertical stack, so the tools used most frequently in kitchen design would logically be placed in a similar order.

62. The claim here unfortunately simply does not fit the facts of the case: the top few icons in the vertical toolbar of both 20-20 Design and ProKitchen have to do with drawing walls, something that generally happens once near the start of a design, while the cabinet, windows, doors, etc., icons, which will be used quite frequently in the design, are further down. While frequency of use is a common user-interface design guideline, the order in both 20-20 Design and ProKitchen is clearly not frequency of use.

63. Note also that Mr. Abbott has (indirectly) acknowledged the significance of the user interface (contrary to his position discussed above). Icon order matters because user interface matters, and it matters in software buying decisions.

64. Mr. Abbott’s report (and the other RealView documents) attempt to downplay the similarity of menu designs in the two programs with a variety of means. In the case of the “Save As Image” command found in both programs, we find the “Google argument” and Mr. Abbott’s

comments about file formats and about the utility of being able to save a design as an image. I consider each of these in turn.

65. The Google argument is unconvincing for a variety of reasons, not the least of which is that Google will return any document containing the three words “save as image,” in that order. It of course has no understanding of the text, so it will return documents whose actual text is “That’s nothing I care to save, as image is not important to me,” or documents that say “The program has no save as image command” (and there is in fact at least one of those in the set of documents Google returns). Second, the common occurrence of the phrase in text has no impact on whether it is commonly used in design programs, or more particularly in the kitchen design programs at issue here.

66. Mr. Abbott’s discussion of the phrase is to a large degree irrelevant, as he explains about file formats at some length, emphasizing a difference that has nothing to do with the issue of copying expression. He also claims [at 15] there is “a compelling reason for the ‘File’ menu to contain a ‘Save as Image’ option,” but then does not support this argument at all. He argues that the ability to save the image as a file matters, but never addresses the issue of expression, i.e., the selection and arrangement of the specific menu items at hand.

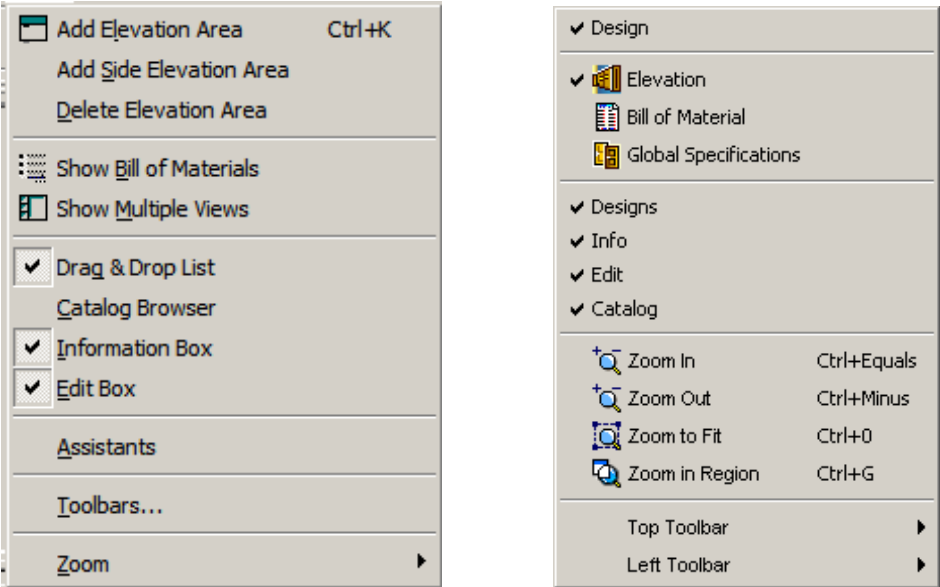
67. None of the RealView documents address the obvious question – how many other comparable programs offer a “Save as Image” menu command? In particular, if we examine the six comparable programs listed at the beginning of this document, we find that *none of them do*. Sweet Home does not have the capability at all, and the other five uniformly use a different phrase and a different menu organization. Every one of those five uses the term “Export” and most employ a multi-level menu structure different from that in 20-20 Design and ProKitchen.

68. Simply put, five of the six programs offer the same functionality, but every one of those expresses it differently, both in terms of the words used and the menu organization. Clearly ProKitchen had alternative choices available for this menu organization and expression: they could have done what was common in the field, or they could have produced a verbatim copy of 20-20 Design's choices. Clearly they did the latter.

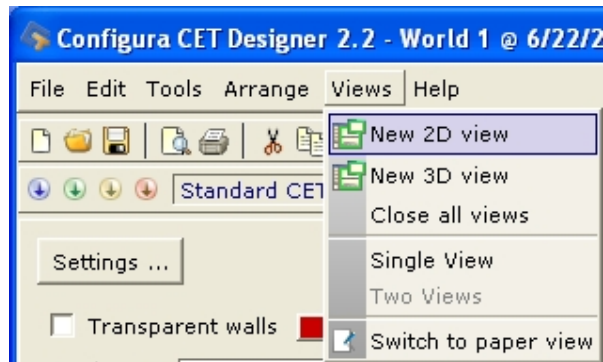
69. Mr. Abbott goes to some effort [at 16-17] to point out differences between the View menus of 20-20 Design and ProKitchen, claiming that "The view menus as displayed are different in look" and that features common between them "are standard in most CAD software because all CAD programs require them."

70. Once again Mr. Abbott fails to make the obvious comparison. For reference I have reproduced below the 20-20 Design and ProKitchen View menus. In addition I have also reproduced the View menus for the other six comparable programs listed at the beginning of this document. Even a cursory examination of the View menus of the other 6 programs demonstrates the opportunity for substantially different selections and organizations of the View menu items.

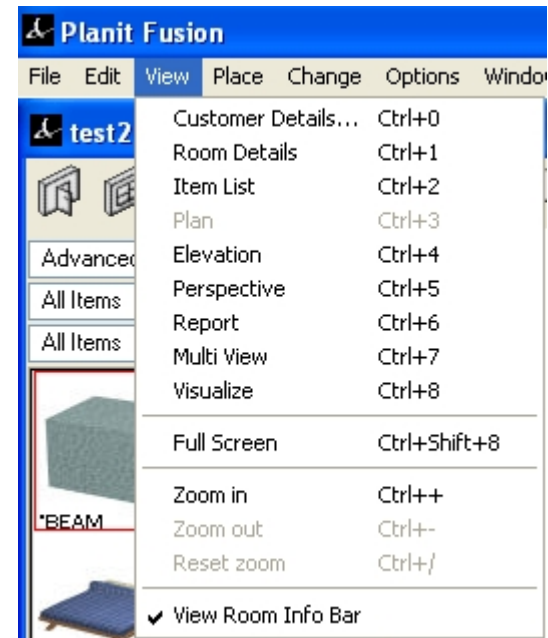
71. Faced with the task of asking which two menus are most alike, there is no contest – 20-20 Design and ProKitchen are clearly far more similar to each other than any of the others, down even to the expressive element of the color schemes and type fonts used (note how different all the other programs are in this respect, among others).



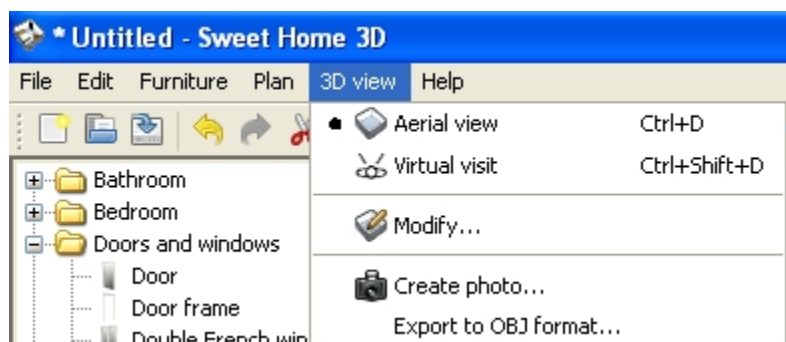
View Menus: 20-20 Design (left) and ProKitchen (right)



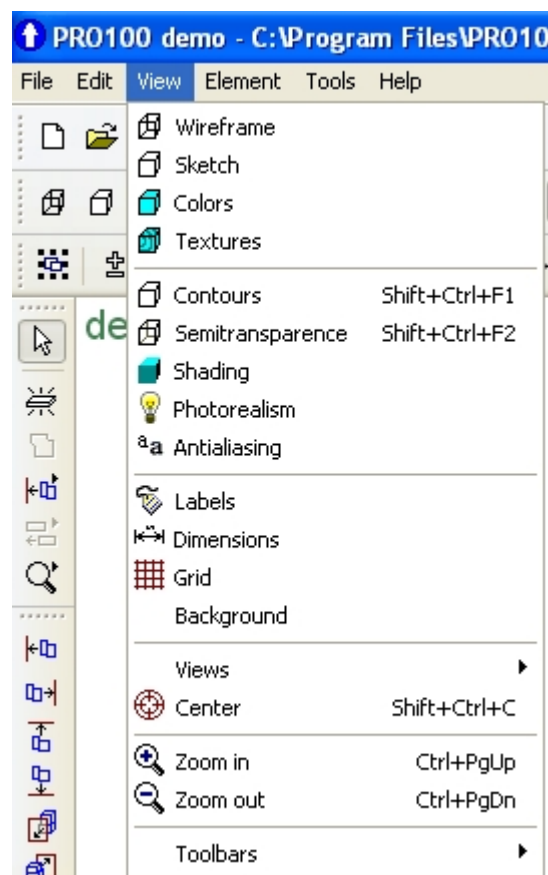
Configura CET Views Menu



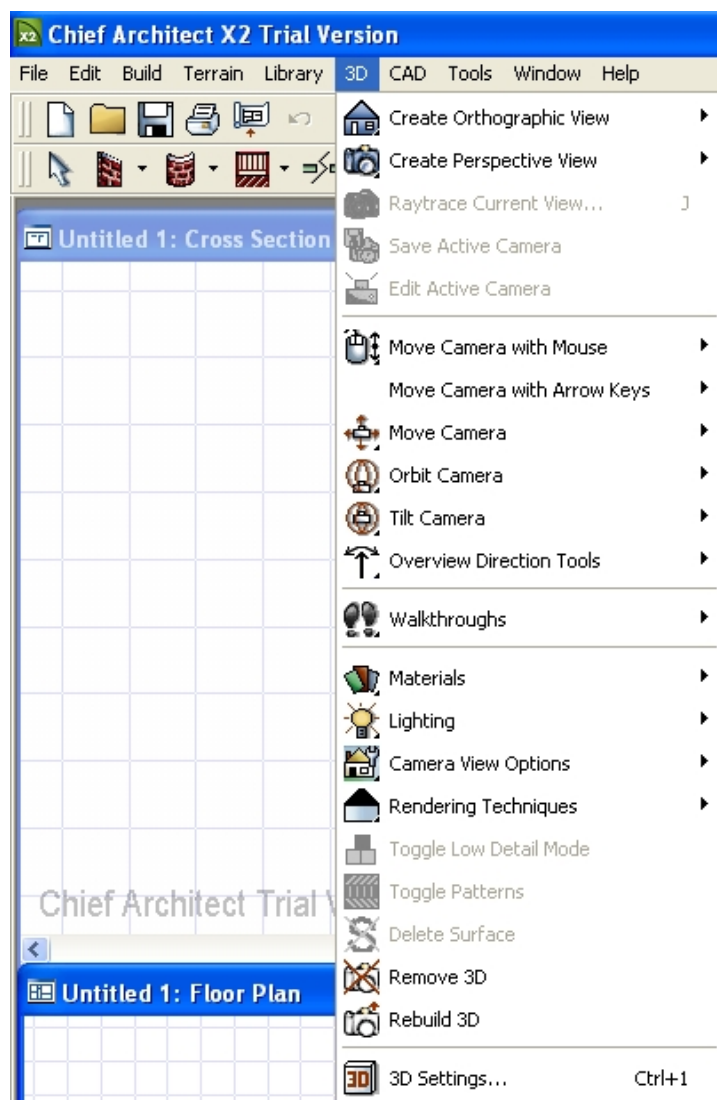
Planit Fusion View Menu



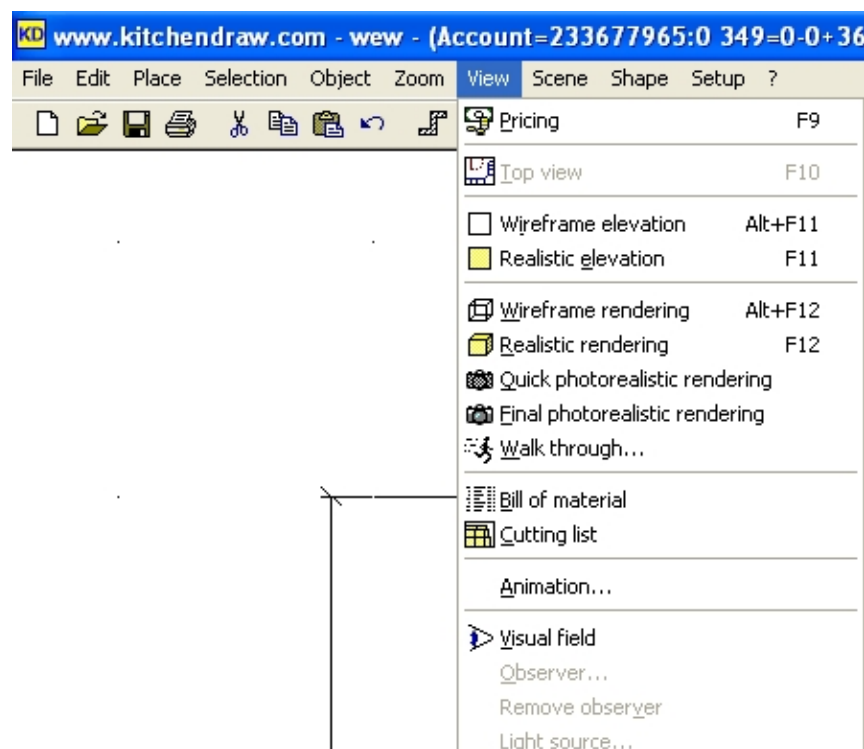
Sweet Home 3D View Menu (the closest match)



Pro100 View Menu



Chief Architect 3D Menu (the closest match)



KitchenDraw View Menu

72. The examples also demonstrate that the claim “all CAD programs require them” is clearly false, at least if the claim is that “all CAD programs require them in the View menu.” If the claim is simply that all CAD programs require the feature *somewhere* in their design, then the argument being made is about functionality and is irrelevant to the question of similarity in expression, selection and arrangement of the menu items, i.e., irrelevant to the issue at hand.

73. Finally, Mr. Abbott’s report (and the other RealView documents) argue against the claim in my report that the expression of the wall drawing process is very similar. They do this largely by ignoring the claim I actually made, generalizing it so they can argue against the straw man. Mr. Abbott, for example, claims [at 26] that “The method of placing a wall in 20-20 Design and ProKitchen is similar to that of many CAD programs that allow entities to be sized by either moving the mouse or typing in a value.”

74. He begins by describing how other programs express wall drawing, indicating [at 26, emphasis added] that:

AutoCAD permits this by *tabbing* between the length and angle option as a line is placed.

SolidWorks provides a *floating dialog box* to change a dimension.

Revit Architecture permits the length to be *entered in a floating window*.

75. As the emphasized words above show, his examples immediately prove how much other programs *differ* in their expression of wall drawing. None of them express it using the idiosyncratic left mouse/right mouse combination. Surely ProKitchen had available these options as other choices, yet they chose a combination that is almost exactly what 20-20 Design did.

76. The Zeldin affidavit (Exh D pg 53) is correct in indicating that my report has a small error in it, where I claimed that the sequence of mouse clicks is perfectly identical. It is not; ProKitchen allows changing both wall length and wall angle after a right-mouse click. The fact

that this is the only difference between the two is revealing, especially given the list provided by Mr. Abbott about how other programs express the idea.

77. We can also look at the other six comparable programs to see how they express the idea of drawing walls:

- a) Planit Fusion Live: The wall layout wizard starts before the design begins, so the room dimension and basic wall layout is set without use of the mouse. Additional walls can be drawn by selecting the wall type (e.g., height) from the wall palette, select the appropriate Item tool (e.g., polyadd item tool), left click starts wall, drag mouse (with button up or down, doesn't matter), wall length and direction follow the mouse, left or right click starts new wall segment.
- b) Sweet Home: select Add Wall (in the Plan menu, or via icon), left click starts wall, length and direction follow the mouse (whether mouse is held down or not), left click starts new segment, double left click ends wall. Right click produces response unrelated to walls.
- c) Pro100: The room dimension and basic wall layout is set at the very beginning of the design so the walls are for the most part taken care of without the mouse. Additional walls can be added by selecting walls in the catalog, and dragging a wall into the design. Walls come in single segments of default length, width and height; these can be changed with the mouse, but each wall segment must be added independently.
- d) KitchenDraw: The room dimension and basic wall layout is set at the very beginning of the design so the walls are for the most part taken care of without the mouse. To add additional walls later, select the Walls icon, left click starts wall,

length and direction follow mouse, left click ends segment, starts another, double-left click ends wall, right click erases previous segment.

- e) Chief Architect: Click on Add Wall icon, left click and hold to start drawing, mouse-up ends the segment, wall length and angle follows the mouse. Right click produces response unrelated to wall drawing.
- f) Configura CET: select wall from palette, left click and drag to draw wall, length and direction follow mouse, mouse-up ends wall segment; right click has no effect.

78. Both 20-20 Design and ProKitchen also offer the option to “extrude” a wall, by which they mean place a U-shaped indentation in the wall. They offer not just the same functionality, but the same means of expressing it: a command called “extrude,” in a context menu. Once again, none of the six competing programs offers this expression (nor for that matter, the functionality). While Chief Architect has an Extrude command, it expresses the far more traditional CAD-related meaning, of taking a 2 dimensional face and moving perpendicular to the face to create a solid.

79. Finally, consider the means chosen in 20-20 Design to express the idea of terminating the drawing of a continuous sequence of wall segments. As both the right click and left click of the mouse already express something, the program offers the quite standard option of hitting the escape key (ESC) to terminate wall drawing (as does ProKitchen). Interestingly, 20-20 Design offers a second means of expressing this: the user can left click behind the most recently drawn wall segment. This is interesting for our current purposes because (a) it apparently does not appear anywhere in the 20-20 Design documentation, (b) it apparently does not appear anywhere in the ProKitchen documentation, and yet (c) ProKitchen offers exactly the

same, rather obscure, means of expressing the termination of a wall sequence (which does not work in any of the other programs mentioned). This suggests the attention to detail exercised by ProKitchen programmers in copying elements of 20-20 Design.

80. As is clear, there is a wide variety of ways to express the idea of drawing a wall, with many variations even within the six programs for kitchen design. Clearly ProKitchen had available a variety of other choices it could have made, yet it chose almost exactly what 20-20 Design had chosen.

81. This set of examples of shortcomings in the RealView documents is illustrative, rather than exhaustive, but provides an indication of the failure of their arguments.

III D. *The Abstraction, Filtration, Comparison Test*

82. There are several ways in which the programs mentioned in this case may usefully be abstracted. One way concerns the category of the program (Fig 1). At the most abstract level these are all computer programs (of course). More specifically they are all CAD programs; more specifically they are architectural CAD (rather than, e.g., mechanical CAD, like e.g., SolidWorks or AutoCAD, or other CAD domains); more specifically they are kitchen design CAD; and more specifically yet they may be kitchen design CAD programs intended for CAD professionals, or they may be kitchen design CAD programs intended for kitchen sales personnel. In this last category we find 20-20 Design, ProKitchen, and five of the six other programs listed at the beginning of this document (Sweet Home 3D, as noted, is oriented more toward the homeowner.)

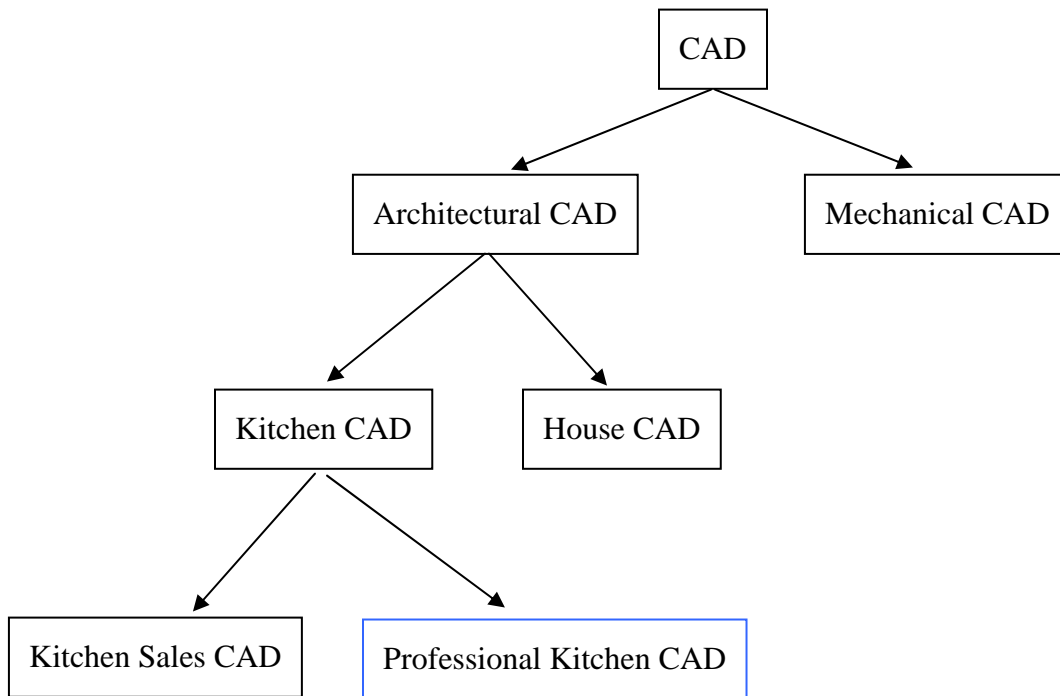


Figure 1: Abstraction hierarchy program category

83. We can also describe an abstraction hierarchy for the interface. At the highest level there is the abstract notion of an interface, then a graphical user interface (GUI) vs. a character-oriented interface. Within GUIs there are interfaces that work strictly from menus selected with the keyboard, those that work from the keyboard for text and use the mouse as a pointing/selecting device, and those that use the keyboard for text and use the mouse as a pointing/selecting device and as a drawing device. (Different combinations of these are of course possible as well.) All these GUI-oriented programs can be further specified as using a single window at once, or (as is more common), permit multiple windows.

84. Finally, we can describe kitchen design programs themselves in several levels of abstraction. At the top is the overall task of kitchen design, at the next level we have the various sub-tasks that go into kitchen design (e.g., create walls, add cabinets, add appliances, add

countertops, etc.). At the next level we can break down creating walls, for example, into indicating their location, extent, angular orientation, etc.

85. As much of the discussion in my original report, this report, and the RealView documents concerns similarity and filtering, I will not repeat those details here. Instead I will summarize my opinion by saying that there are substantial similarities in 20-20 Design and ProKitchen, and that the arguments for filtering presented in the RealView documents appear to me to be unconvincing, for the numerous reasons enumerated above.

86. Given the numerous similarities between the two programs that survive filtration, and their level of detail (e.g., the selection and arrangement of displays, selection and arrangement of commands, details of wall drawing, etc.) it seems clear to me that there is substantial unfiltered expression in 20-20 Design that has been copied by ProKitchen.

IV. CONCLUSIONS

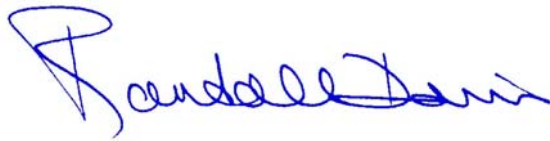
87. In summary, the RealView documents are unsuccessful in their attempts to argue that the similarities noted in my report of May 5th should be filtered. Their arguments fail for a number of reasons, including because they create their own, generalized straw-man versions of the similarities I cited, because they ignore the expressive aspects of 20-20 Design, because they must find individual elements of 20-20 Design in numerous different programs, but can never cite a program other than ProKitchen that offers the same selection and arrangement of elements, because they offer arguments that are irrelevant to the issue at hand, and because they apply what appear to be novel (and unsubstantiated) criteria for filtration.

88. The program examples they cite are for the most part unsuccessful because they concern their own generalizations of the similarities, not the specific similarities I pointed out,

and because they rigorously avoid comparing to programs intended for the same market, namely computer aided sales for kitchen designers.

89. As a result of the failure of the filtration proposed in the RealView documents, an abstraction, filtration, comparison analysis of the two programs demonstrates substantial similarities at the level of detailed expression in the two programs.

90. Finally, the detailed expression common to the two programs that is discussed above goes much beyond menu commands, and concerns among other things, issue of choice, selection, and arrangement of expressive elements of the screen layout, display, and appearance.

A handwritten signature in blue ink, appearing to read "Randall Davis", is positioned above a horizontal line.

Randall Davis